

**FIRST AND SECOND MANAGER COMPONENTS THAT PERFORM ONE OR MORE FIRST AND SECOND MANAGEMENT OPERATIONS ON A SOFTWARE AND/OR HARDWARE ENTITY**

**TECHNICAL FIELD**

5           The invention relates generally to software and/or hardware and more particularly to software and/or hardware entity management.

**BACKGROUND**

A manager component in one example controls one or more software and/or hardware entities. For example, the manager component is responsible for starting, stopping, 10 initializing, and monitoring the entity, detecting failures of the entity, recovering the entity, and propagating state changes about the entity. In one example, the entity operates in a high availability domain. So, the manager component comprises high availability services software. In another example, the entity operates outside of the high availability domain. So, the entity does not interact with the high availability services software.

15           The software and/or hardware entity in one example comprises one or more software and/or hardware components (e.g., software and/or hardware sub-entities). The manager component has control of all management operations on the software and/or hardware entity and the software and/or hardware components. For example, the manager component maintains autonomous control of the software and/or hardware entity.

20           Some circumstances exist where autonomous control is either undesirable or impossible. In one example, upon introduction of new high availability services software into a legacy system, the new high availability services software must interoperate with the old high availability services software already in the legacy system. As one shortcoming, the autonomous control required by the new high availability services software may prevent the 25 old high availability services software from controlling any portion of the software and/or

hardware entity. In another example, the software and/or hardware entity may need a certain level of application-specific control over the management of some critical portion of the software and/or hardware entity. As another shortcoming, the autonomous control by the high availability services software may not provide the software and/or hardware entity with the desired level of application-specific control.

Thus, a need exists for a manager component that provides a software and/or hardware entity with a desired level of application-specific control.

### **SUMMARY**

The invention in one embodiment encompasses an apparatus. The apparatus comprises a first manager component that performs one or more first management operations on a software and/or hardware entity and a second manager component that performs one or more second management operations on the software and/or hardware entity.

Yet another embodiment of the invention encompasses a method. A software and/or hardware entity is configured for partial control by a first manager component and partial control by a second manager component.

Still another embodiment of the invention encompasses an article. The article comprises one or more computer-readable signal-bearing media. The article comprises means in the one or more media for configuring a software and/or hardware entity for partial control by a first manager component and partial control by a second manager component.

### **DESCRIPTION OF THE DRAWING**

Features of exemplary implementations of the invention will become apparent from the description, the claims, and the accompanying drawing in which:

FIG. 1 is a representation of an exemplary implementation of an apparatus that comprises one or more software and/or hardware entities and a plurality of manager components.

### **DETAILED DESCRIPTION**

5 Turning to FIG. 1, an apparatus 100 in one example comprises one or more software and/or hardware entities 102 and a plurality of manager components 104 and 106. The manager components 104 and 106 in one example share control of the software and/or hardware entity 102. For example, the manager components 104 and 106 together are responsible for starting, stopping, initializing, and monitoring the software and/or hardware  
10 entity 102, detecting failures of the software and/or hardware entity 102, recovering the software and/or hardware entity 102, propagating state changes about the software and/or hardware entity 102, and the like.

The software and/or hardware entity 102 in one example provides functionality for a computer system. In one example, the software and/or hardware entity 102 is responsible for  
15 setup and/or teardown of telecommunication connections in a telecommunication network. For example, the manager components 104 and 106 and the software and/or hardware entity 102 may cooperate to setup and/or teardown communication sessions. The software and/or hardware entity 102 in one example comprises one or more software and/or hardware components 108 and 110. For example, the software and/or hardware entity 102 may  
20 comprise tens, hundreds, or thousands of software and/or hardware components. The software and/or hardware components 108 and 110 represent software and/or hardware sub-entities of the software and/or hardware entity 102. For example, the software and/or hardware components 108 and 110 represent logical blocks of software and/or hardware of the software and/or hardware entity 102. The software and/or hardware components 108 and

110 in one example are developed independently and then put together within the software and/or hardware entity 102.

The software and/or hardware entity 102 and the manager components 104 and 106 are communicatively coupled to allow exchange of management commands and responses.

5 Also, the manager components 104 and 106 are communicatively coupled to allow exchange of requests and notifications about management operations. The manager components 104 and 106 share responsibility for controlling the software and/or hardware entity 102. For example, each of the manager components 104 and 106 have partial control of the software and/or hardware entity 102. The software and/or hardware entity 102 comprises a hybrid  
10 software and/or hardware entity that is not under autonomous control by a single manager. In one example, two manager components each have partial control of the software and/or hardware entity 102. In another example, three or more manager components each have partial control of the software and/or hardware entity 102.

The manager components 104 and 106 divide the responsibility for management  
15 operations on the software and/or hardware components 108 and 110. In one example, both of the manager components 104 and 106 may perform management operations on the same ones of the software and/or hardware components 108 and 110. For example, the manager component 104 may perform one or more first management operations on the software and/or hardware component 108 and the manager component 106 may perform one or more second  
20 management operations on the software and/or hardware component 110. In another example, each of the manager components 104 and 106 are responsible for different ones of the software and/or hardware components 108 and 110. For example, the software and/or hardware components 108 and 110 may comprise one or more first software and/or hardware components 108 and one or more second software and/or hardware components 110. The  
25 manager component 104 in one example controls the one or more first software and/or

hardware components 108. The manager component 106 in one example controls the one or more second software and/or hardware components 110.

The manager component 104 in one example performs one or more first management operations on the software and/or hardware entity 102. The manager component 106 in one example performs one or more second management operations on the software and/or hardware entity 102. To share management responsibility between the manager components 104 and 106, individual responsibilities are defined for each of the manager components 104 and 106 regarding the software and/or hardware entity 102. To keep each of the manager components 104 and 106 apprised of the activities of the other of the manager components 104 and 106, the manager components 104 and 106 exchange notifications. The manager components 104 and 106 may exchange notifications that indicate management operations that are about to take place or have already taken place. Also, upon detection by the management component 104 of an event associated with the software and/or hardware entity 102, the manager component 104 may send a notification to the manager component 106 to indicate occurrence of the event. Upon detection by the management component 106 of an event associated with the software and/or hardware entity 102, the manager component 106 may send a notification to the manager component 104 to indicate occurrence of the event.

Sending the notifications between the manager components 104 and 106 helps the manager components 104 and 106 determine the current state of the software and/or hardware entity 102 and the software and/or hardware components 108 and 110. Knowing the current state of the software and/or hardware entity 102 and the software and/or hardware components 108 and 110 allows the manager components 104 and 106 to coordinate management operations performed on the software and/or hardware entity 102 and the software and/or hardware components 108 and 110 so that the operations occur in a proper sequence. For example, the software and/or hardware entity 102 must be started before being

stopped. So, the manager components 104 and 106 understand the management operation ordering required and coordinate to ensure that management operations are properly sequenced.

The manager components 104 and 106 in one example are responsible for  
5 determining their own responsibilities with regards to managing the software and/or hardware entity 102. In one example, the responsibilities are statically defined. For example, the manager components 104 and 106 access a configuration file that provides each of the manager components 104 and 106 with a list of the responsibilities. In another example, the manager components 104 and 106 dynamically negotiate the responsibilities. For example,  
10 the manager components 104 and 106 each agree to specific management operations or control of a specific portion of the software and/or hardware entity 102. The manager component 104 may send a notification to the manager component 106 to request specific responsibilities and the manager component 106 may also send a notification to the manager component 104 to request specific responsibilities.

15 One of the manager components 104 and 106 may send a request to another of the manager components 104 and 106 to trigger a management operation. For example, the manager component 104 may send a request to the manager component 106 to cause the manager component 106 to perform a management operation on the software and/or hardware entity 102. The manager component 106 in one example accepts the request,  
20 performs the management operation, and sends an acknowledgement or error indication to the manager component 104.

One or more of the manager components 104 and 106 in one example comprise high availability system software operating in a high availability domain. For example, the manager component 106 may operate within the high availability domain and the manager  
25 component 104 and the software and/or hardware entity 102 may operate outside of the high

availability domain. The software and/or hardware components 108 and 110 of the software and/or hardware entity 102 operate outside of the high availability domain. However, the software and/or hardware entity 102 and the software and/or hardware components 108 and 110 may interact with the high availability system software in the high availability domain.

5 So, the software and/or hardware entity 102 may exist outside of the high availability domain while still employing one or more functions of the high availability services software. The software and/or hardware entity 102 may allow connection with the management component 106 (e.g., the high availability services software) and allow connection with the management component 104 to prevent autonomous control of the software and/or hardware entity 102 by  
10 one of the management components 104 and 106, such as, the high availability services software. For example, the high availability services software is able to peer with the management component 104 to cooperatively manage the software and/or hardware entity 102 not under the exclusive control of the high availability services software.

The software and/or hardware entity 102 in one example may desire a certain level of  
15 application-specific control over the management of some critical portion of the software and/or hardware entity 102. The autonomous control by the high availability services software may not provide the software and/or hardware entity 102 with the desired level of application-specific control. To gain the desired level of application-specific control while retaining some desired management operation of the high availability services software, a  
20 first portion of the management responsibilities of the software and/or hardware entity 102 is controlled by the management component 104 and a second portion of the management responsibilities of the software and/or hardware entity 102 is controlled by the high availability services software. The partial control of the software and/or hardware entity 102 by the management component 104 may provide the software and/or hardware entity 102  
25 with the desired level of application-specific control.

An illustrative description of one exemplary operation of the apparatus 100 is now presented, for explanatory purposes. A message flow 120 represents an exemplary startup and initialization of the software and/or hardware entity 102 by the manager components 104 and 106.

5 In one example, the manager component 104 is responsible for startup of the software and/or hardware entity 102. Upon determination to startup the software and/or hardware entity 102, the manager component 104 sends a startup command 122 to the software and/or hardware entity 102. The startup command 122 activates the software and/or hardware entity 102 and the software and/or hardware components 108 and 110. Upon execution of the  
10 startup command 122, the software and/or hardware entity 102 in one example sends a response to the manager component 104 to indicate success of the startup command 122. To provide an indication that the software and/or hardware entity 102 has been started, the manager component 104 sends a startup notification 124 to the manager component 106. The startup notification 124 serves to keep the manager component 106 apprised of the activities  
15 of the manager component 104 and the state of the software and/or hardware entity 102. The manager component 106 waits for the startup notification 124 before initiating management operations with the software and/or hardware entity 102.

The manager component 106 in one example is responsible for one or more management operations for the software and/or hardware entity 102. The manager  
20 component 106 in one example is responsible for initializing the software and/or hardware component 110. Upon receipt of the startup notification 124, the manager component 106 sends an initialization command 126 to the software and/or hardware component 110. Upon execution of the initialization command 126, the software and/or hardware component 110 in one example sends a response to the manager component 106 to indicate success of the  
25 initialization command 126. To provide an indication that the software and/or hardware



component 110 has been initialized, the manager component 106 sends an initialization notification 128 to the manager component 104. The initialization notification 128 serves to keep the manager component 104 apprised of the activities of the manager component 106 and the state of the software and/or hardware entity 102.

5           The message flow 120 illustrates startup and initiation of the software and/or hardware entity 102, however, the message flow 120 is analogous to other management operations. For example, the command, response, notification message sequence between the software and/or hardware entity 102 and the manager components 104 and 106 illustrated in the message flow 120 is analogous to a message sequence for other management operations,  
10 such as, stopping the software and/or hardware entity 102, monitoring the software and/or hardware entity 102, detecting failures of the software and/or hardware entity 102, reporting the failures of the software and/or hardware entity 102, and the like.

          In one example, the management component 104 detects a failure in the software and/or hardware component 108. The management component 104 sends a failure  
15 notification to the management component 106. Upon receipt of the failure notification, the management component 106 may initiate a management operation on the software and/or hardware entity 102 or another software and/or hardware entity based on the failure notification.

          The apparatus 100 in one example comprises a plurality of components such as  
20 electronic components, computer hardware components, and/or computer software components. A number of such components can be combined or divided in the apparatus 100. An exemplary component of the apparatus 100 employs and/or comprises a set and/or series of computer instructions written in or implemented with any of a number of programming languages, as will be appreciated by those skilled in the art.

The apparatus 100 in one example employs one or more computer-readable signal-bearing media. One example of a computer-readable signal-bearing medium for the apparatus 100 comprises one or more instances of the recordable data storage medium 150 of one or more of the manager components 104 and 106 and the software and/or hardware entities 102. For example, the recordable data storage medium 150 comprises one or more of a magnetic, electrical, optical, biological, and atomic data storage medium. In another example, a computer-readable signal-bearing medium for the apparatus 100 comprises a modulated carrier signal transmitted over a network comprising or coupled with the apparatus 100, for instance, one or more of a telephone network, a local area network (“LAN”), the internet, and a wireless network.

The steps or operations described herein are just exemplary. There may be many variations to these steps or operations without departing from the spirit of the invention. For instance, the steps may be performed in a differing order, or steps may be added, deleted, or modified.

Although exemplary implementations of the invention have been depicted and described in detail herein, it will be apparent to those skilled in the relevant art that various modifications, additions, substitutions, and the like can be made without departing from the spirit of the invention and these are therefore considered to be within the scope of the invention as defined in the following claims.